



Fedora Directory Server

FUDCon III – London, 2005

Overview of LDAP

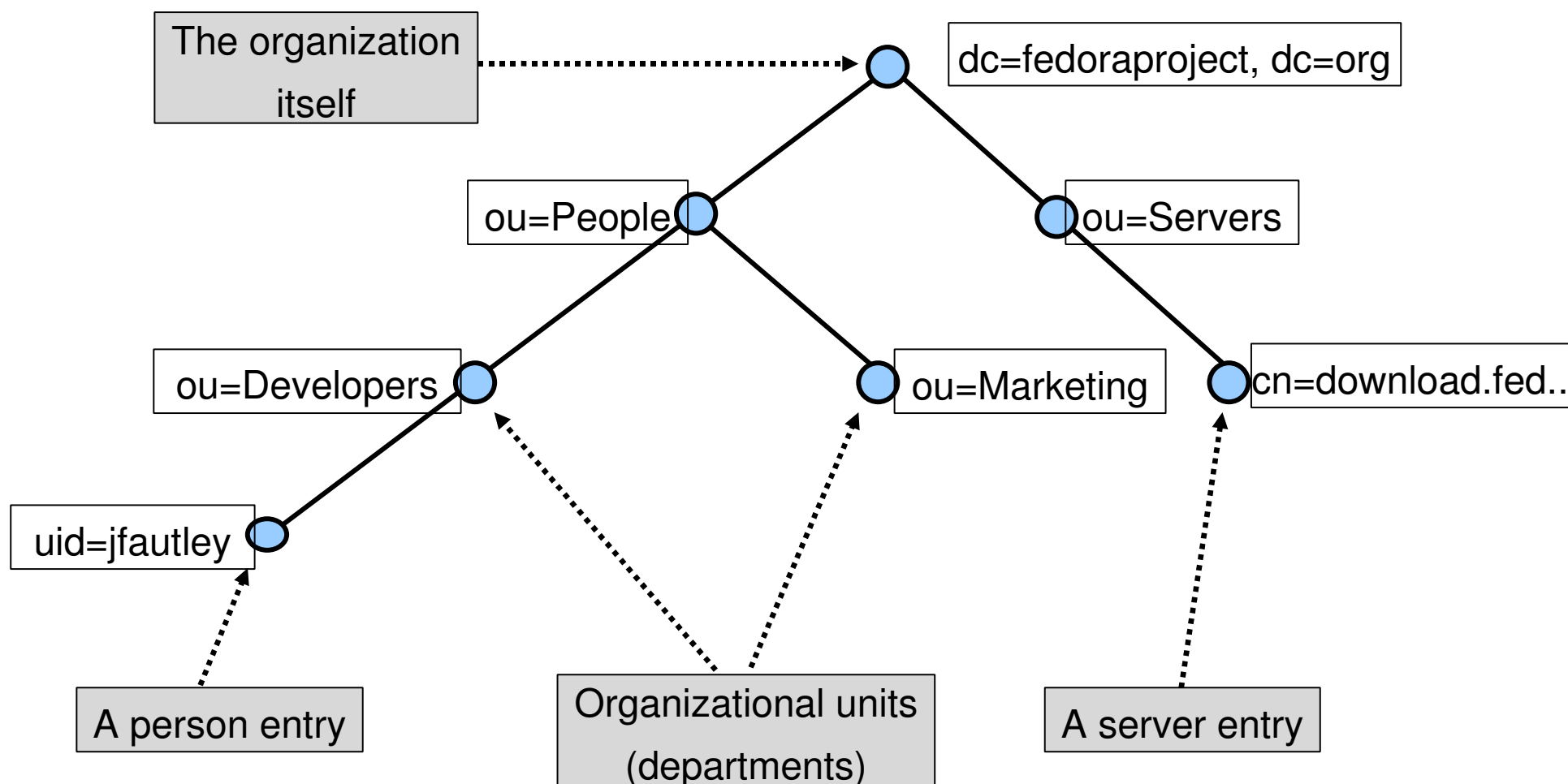
What Is LDAP?

- **L**ightweight **D**irectory **A**ccess **P**rotocol
- Widely supported, standard protocol, up to version 3
- Began at the University of Michigan in the early 1990s to provide "lightweight" access to X.500 directories using ordinary TCP/IP
- Defined by many IETF RFCs
- Ongoing work in the IETF LDAP working groups
- Implemented in many different directory products

Characteristics of a Directory Service

- Designed to have a high read-to-write ratio
- Provides scoped (subtree) search
- Allows partitioning, distribution, and delegation of control
 - Hierarchical naming
 - Loosely consistent replication
 - Sophisticated authentication & access control
- **Benefits for the Enterprise:**
 - Centralizes system administration
 - Enables multi-vendor solutions
 - Reduces cost of ownership
 - Saves development time for user and group management

Sample Directory Information Tree (DIT)



Differences Between LDAP Directory and Relational DB

■ Relational Database:

- High update performance
- Relational data model; no hierarchy
- Tight replication consistency – two phase commit

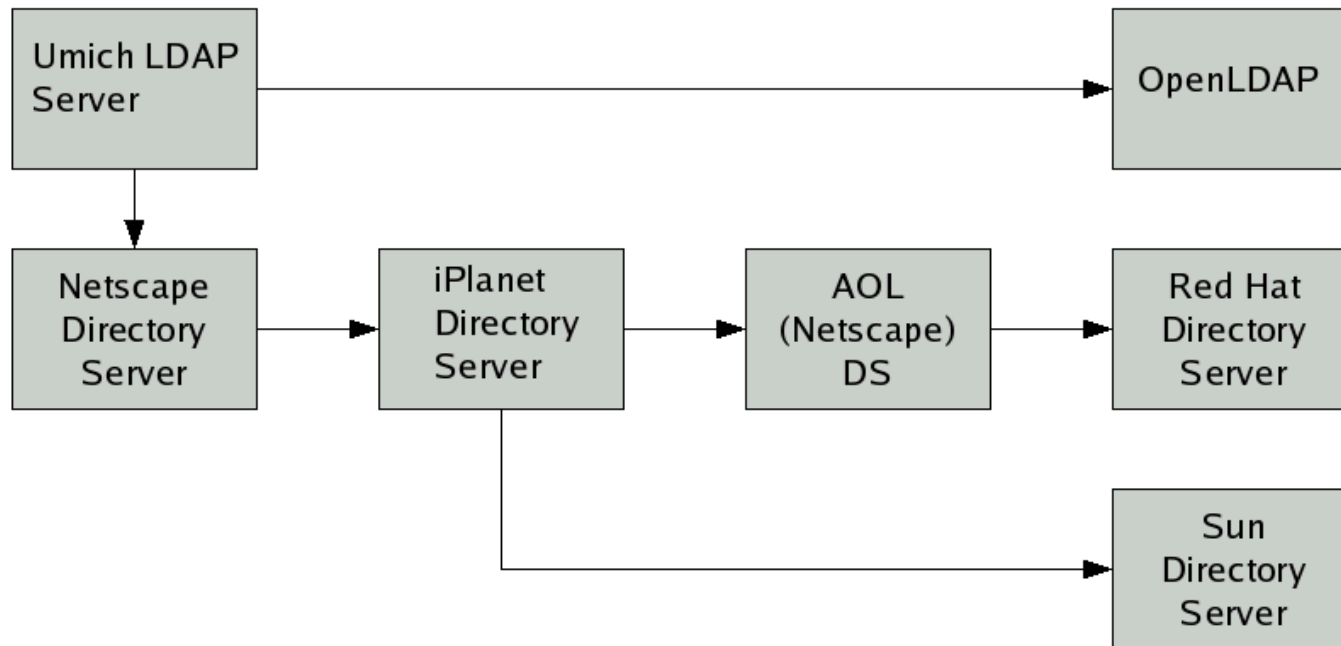
■ LDAP Directory:

- High search performance
- Scope matters – hierarchical data model; no relational join
- Loose replication consistency, wide data availability



Fedora Directory Server

Code History



Fedora Directory Server

- Red Hat purchased assets from AOL which included
 - Netscape Directory Server
 - Netscape Certificate System
 - Mail and Calendar applications
 - People
 - Customers
- Directory Server allows us to glue a lot of technologies together in one location
- Red Hat provided the Directory Server to the Open Source community
- No other open source enterprise-ready directory servers out there
- Netscape Directory Server has customer stories to tell and has long history

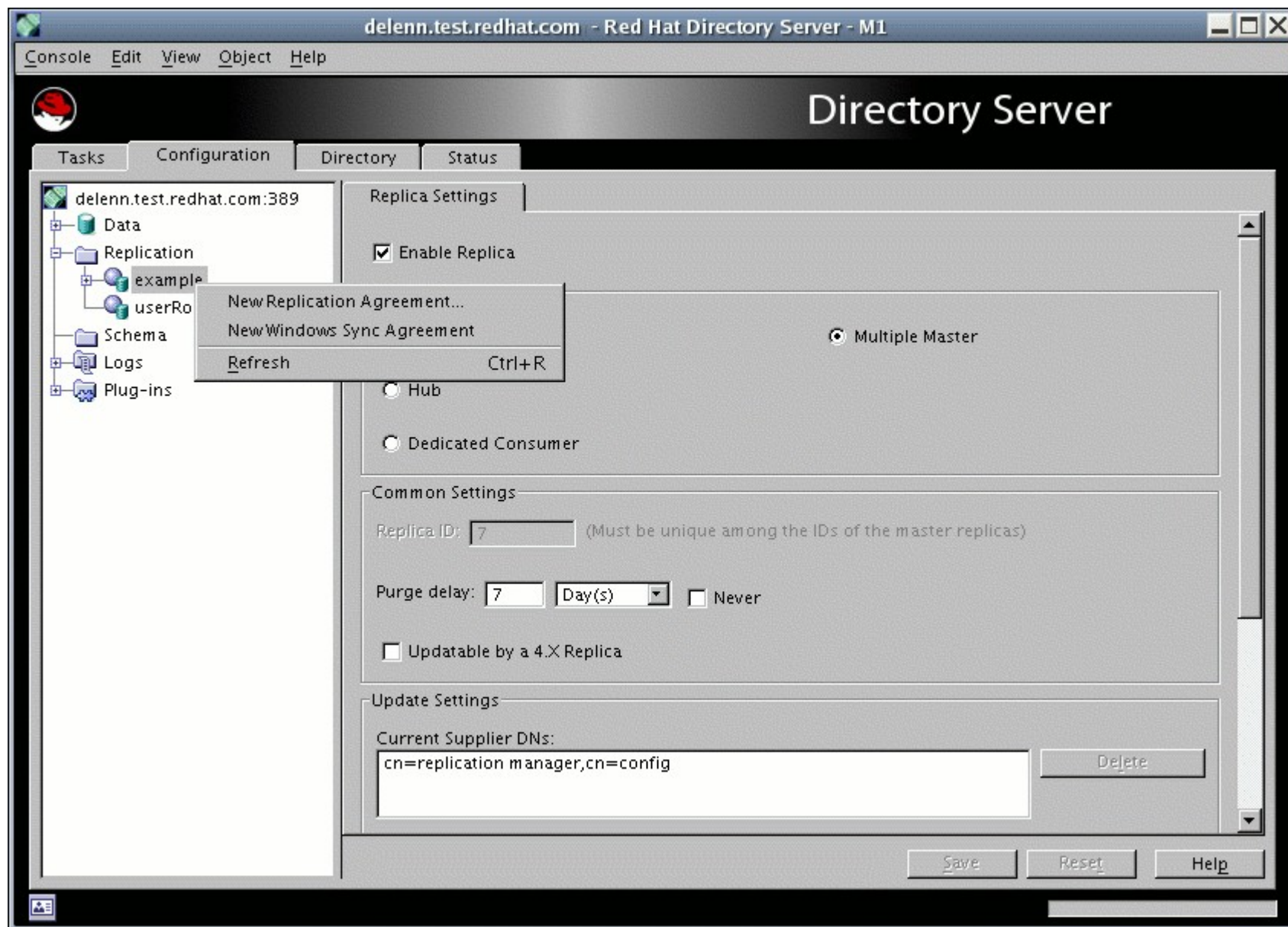
Fedora Directory Server: Scalable Identity

- **LDAP-based authentication** (“who are you”):
 - Widely supported; OS access through NIS or PAM “gateway”
 - Supports Kerberos via SASL GSS-API mechanism
 - Integrated support for X.509 certificates
 - Supports databases, legacy systems via plug-in API
- **Fine-grained access control** (“what can you do”)
 - Using external criteria
 - e.g. type of connection, day of week/time, hostname/IP
 - Using groups (“engineering”) and roles (“managers”)
 - Controls access to services, devices, other resources
- **High availability and scalability** through Multi-Master Replication, chaining, and distribution

Key Features: Flexible Administration

- Many tasks possible without downtime, e.g.:
 - Change server and database configuration
 - Bulk-load data, export data, back up database
 - Add new/change schema, create new indexes
- LDAP used for configuration and monitoring
 - Configuration exposed as a set of LDAP entries
 - Real-time status and statistics available over LDAP
 - Configuration files are LDIF
 - File format same as LDAP format
 - Can also use SNMP for monitoring
- Administration can be done from the command line or a GUI console

Console Interface



Key Features: Performance and Scalability

- High-speed search and retrieval
 - Thousands of reads per second per server
 - Hundreds of writes per second per server
 - Reflects years of experience with very large deployments.
- High-performance data store
 - Designed for 10M+ entries on average
 - Maximum capacity in the terabytes (up to OS limit)
- Data distribution for scalability
 - Server farm distributes entries; server farm to store "buckets"
 - Distribution schemes are plug-ins
 - Scalability to hundreds of millions of entries exists

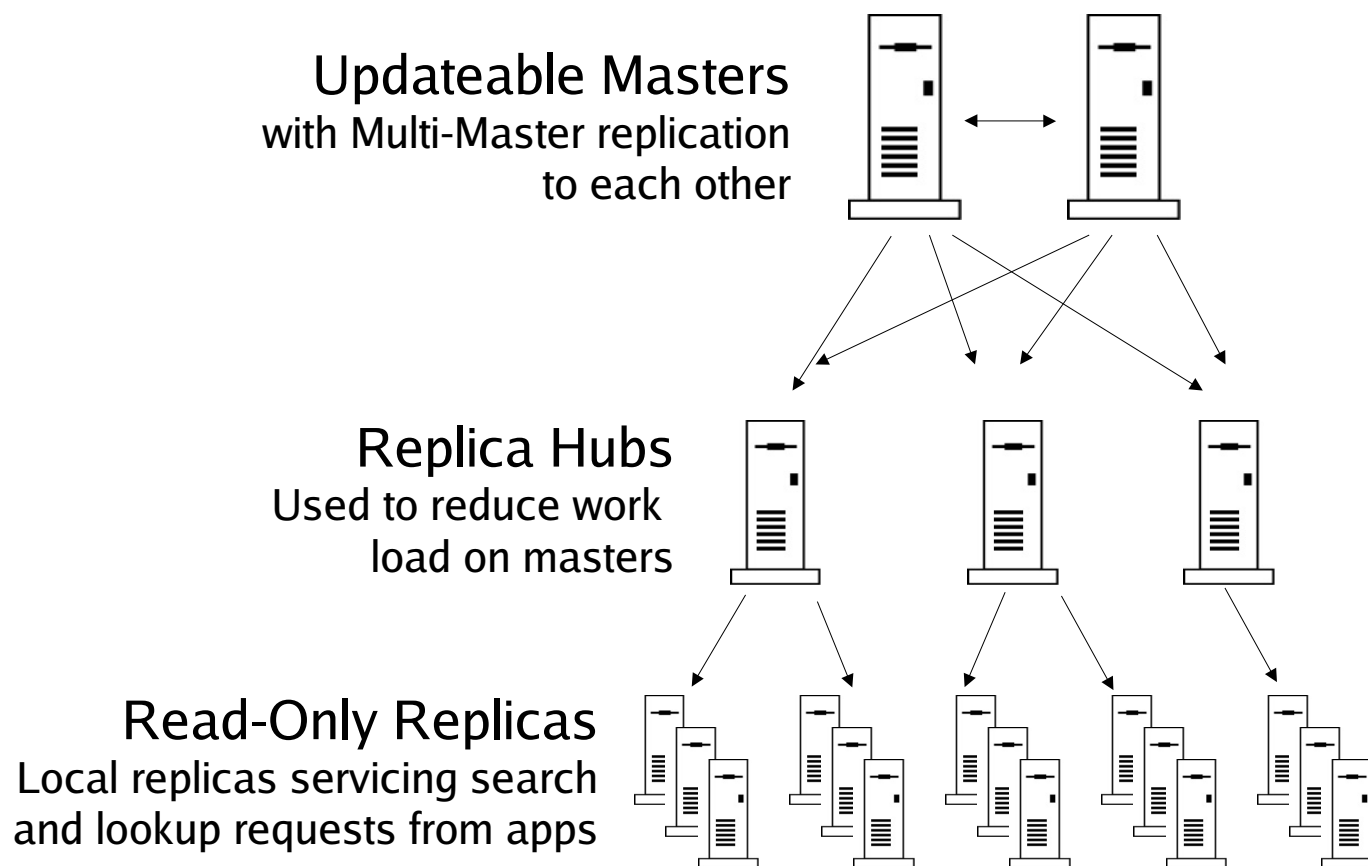
Key Features: Reliability and Availability

- High availability through multi-master replication
 - Simultaneous update with conflict resolution
 - Write availability— no single point of write failure
 - Read availability— put data close to the application
 - Data redundancy for failover, load balancing
- When you make a change to one server, you don't have to wait for change to propagate to all servers
- Fault tolerance via database transaction logging with automated recover

What Is Multi-Master Replication?

- Master copies reside on multiple servers
- Masters can be situated in different data centers, different geographic areas
- Changes to data can be made to closest server, and are then propagated to the other masters
- Failover ensures continuous service
- Automatic time-based conflict resolution
- Not appropriate for every deployment—adds some complexity
- Development requires extensive testing

Key Elements of Multi-Master Replication



Deployment Scenarios

- Very high availability requirements
- Multiple masters running in different geographic locations
- Replication
 - Supplier/Consumer or Multi-Master
 - Over “slow” links
 - Fractional Replication
- Chaining
- Performance
 - Indexes
 - Multiple front end search servers
- Online Backups

Key Features: Access Control

- Access can be controlled down to the attribute value level if desired
 - Very fine-grained control
- Hierarchical (scope matters)
 - Lower levels inherit access control from higher levels
- Access control info stored with data on server
- Access controls based on user or group membership, IP or domain name, time of day, and many other criteria

Key Features: Security

- Multiple authentication mechanisms
 - Userid/passwd, Digest MD5
 - User certificate (X.509)
 - Kerberos authentication via SASL/GSSAPI
 - Impersonation (proxy) for multi-tier client apps
- Built-in password management
 - Hack protection—login tries, account lockout
 - Crack protection—minimum length, no "trivial" passwords, password history
 - Selectable password storage (clear, crypt, SHA1)
- Secure access using TLS (SSL) and certificates
- Denial of Service protection

Key Features: Plug-in API

- Easily customize and extend Directory Server via the plug-in API (C/C++)
- Categories of plug-ins
 - Pre-operation “filters”
 - Post-operation “triggers”
- **Core features**, such as the data store, access control, replication implemented with the Plug-in API
- Plug-ins can be turned on or off as needed

Roadmap

- Solidify leadership in enterprise directory features. Under consideration:
 - x86 64-bit support
 - MS Active Directory improvements
 - Better manage resource consumption
 - Expose High Availability features of SleepyCat backend database
 - Virtual directory features
- Layered products built on core Directory capabilities, such as
 - Improved desktop applications (phonebook, organisation chart)
 - Identity management user portal
 - Identity management/policy server

Questions?



<http://fedora.redhat.com/>

<http://www.fedoraproject.org/>